DETAILED ACTION

1. Claims 1 – 27 are pending in the application. By this examiner's amendment, claims 4, 5, 10, 11, 16, 17, 19, 20 and 27 are cancelled and claims 1, 2, 3, 6-9, 12-15, 18 and 21-26 are amended.

INTERVIEW SUMMARY

2. During a telephone interview with Mr. Robert Voigt on May 21, 2008, examiner indicated that claims 1, 7, 13 and 27 would be rejected over U.S. Patent No. 5,524,202 and U.S. Patent No. 5,548,759. However, the claims would be allowable if amended to recite "converting a read-only database file into a shared library with an executable code segment, wherein the shared library includes a reference pointer to data residing in the executable code segment;" "loading at least a portion of the converted read-only database file into a main memory by executing the shared library;" "obtaining said reference pointer by an application;" and "accessing said at least a portion of the converted read-only database file stored in the main memory based upon said reference pointer by the application." In addition, claims 4, 5, 10, 11, 16, 17, 19, 20 and 27 will be cancelled because they are redundant in scope with their respective independent claims. Applicant agreed with the examiner's suggestion and authorized examiner to amend the claims in an examiner's amendment.

EXAMINER'S AMENDMENT

3. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Mr. Robert Voigt on May 21, 2008.

The application has been amended as follows:

a. Cancel claims 4, 5, 10, 11, 16, 17, 19, 20 and 27;

b. Replace claims 1, 2, 3, 6-9, 12-15, 18 and 21-26 with the following:

1. A method comprising the steps of:

converting a read-only database file into a shared library with an executable code segment, wherein the shared library includes a reference pointer to data residing in the executable code segment;

loading at least a portion of the converted read-only database file into a main memory by executing the shared library;

obtaining said reference pointer by an application; and

accessing said at least a portion of the converted read-only database file stored in the main memory based upon said reference pointer by the application.

2. The method as recited in claim 1, wherein the memory is virtual memory associated with the application running in a processor.

3. The method as recited in claim 2, wherein the converted read-only database file is memory-mapped from system storage by an operating system loader.

6. The method as recited in claim 3, wherein the converting step further comprises the step of wrapping the read only database file with executable code.

7. A data processing system comprising:

circuitry for converting a read-only database file into a shared library with an executable code segment, wherein the shared library includes a reference pointer to data residing in the executable code segment;

circuitry for loading at least a portion of the converted read-only database file into a main memory by executing the shared library;

circuitry for obtaining said reference pointer by an application; and

circuitry for accessing said at least a portion of the converted read-only database file stored in the main memory based upon said reference pointer by the application.

8. The data processing system as recited in claim 7, wherein the memory is virtual memory associated with the application running in a processor.

9. The data processing system as recited in claim 7, wherein the converted read-only database file is memory-mapped from system storage by an operating system loader.

12. The data processing system as recited in claim 7, wherein the converting circuitry further comprises circuitry for wrapping the read only database file with executable code.

13. A computer program tool stored on a storage medium, comprising:

program code for converting a read-only database file into a shared library with an executable code segment, wherein the shared library includes a reference pointer to data residing in the executable code segment;

program code for loading at least a portion of the converted read-only database file into a main memory by executing the shared library;

program code for obtaining said reference pointer by an application; and

program code for accessing said at least a portion of the converted read-only database file stored in the main memory based upon said reference pointer by the application.

14. The computer program tool as recited in claim 13, wherein the memory is virtual memory associated with the application running in a processor.

15. The computer program tool as recited in claim 14, wherein the converted read-only database file is memory-mapped from system storage by an operating system loader.

18. The computer program tool as recited in claim 14, wherein the converting program code further comprises program code for wrapping the read only database file with executable code.

21. The method as recited in claim 6, wherein the wrapping step further comprises the step of wrapping the read only file with code headers and records in order to appear to an operating system loader as the executable code.

22. The data processing system as recited in claim 12, wherein the wrapping circuitry wraps the read only file with code headers and records in order to appear to an operating system loader as the executable code.

23. The computer program tool as recite in claim 18, wherein the program code for wrapping wraps the read only file with code headers and records in order to appear to an operating system loader as the executable code.

24. The data processing system as recited in claim 7 wherein the system includes a processor operating in a protected mode.

25. The method as recited in claim 2, wherein the processor operates in a protected mode.

26. The computer program tool as recited in claim 14, wherein the processor operates in a protected mode.

REASONS FOR ALLOWANCE

4. The following is an examiner's statement of reasons for allowance:

The prior art of record, specifically U.S. Patent No. 5,524,202 to Yokohama and U.S. Patent No. 5,548,759 to Lipe, does not expressly teach or render obvious the invention as recited in independent claims 1, 7 and 13.

The prior art teaches converting [col. 5, lines 2 – 19 of Yokohama and col. 9, lines 23 – 38 of Lipe] a read-only file [col. 5, lines 3 – 18 of Yokohama] into an executable file [col. 4, lines 2 – 18 and col. 8, lines 1 – 15 of Yokohama; col. 10, lines 25 – 67], an operating system faulting a page containing a memory segment [col. 5, lines 40 – 52 of Yokohama] wherein the read-only file resides into memory [col. 9, line 65 – col. 10, line 6 of Yokohama], the memory is virtual memory associated with an application running in a processor [col. 5, lines 40 – 52 of Yokohama], and converted file is memory-mapped from system storage by an operating system loader [col. 9, line 65 – col. 10, line 5 of Yokohama].

However, the prior art does not teach converting a read-only database file into a shared library with an executable code segment, wherein the shared library includes a reference pointer to data residing in the executable code segment; obtaining said reference pointer by an application; and accessing said at least a portion of the converted read-only database file stored in the main memory based upon said reference pointer by the application.

In addition, it is not believed to have been within the level of one of ordinary skill in the art at the time of the invention to modify or integrate the invention of the prior art to incorporate the features “converting a read-only database file into a shared library with an executable code segment, wherein the shared library includes a reference pointer to data residing in the executable code segment", "obtaining said reference pointer by an application”, and “accessing said at least a portion of the converted read-only database file stored in the main memory based upon said reference pointer by the application” as recited in the context of claims 1, 7 and 13.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled “Comments on Statement of Reasons for Allowance.”

CONTACT INFORMATION

5. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Li B. Zhen whose telephone number is (571) 272-3768. The examiner can normally be reached on Mon - Fri, 8:30am - 5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on (571)272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Li B. Zhen
Primary Examiner
Art Unit 2194

/Li B. Zhen/
Primary Examiner, Art Unit 2194